Available online at www.sciencedirect.com
ScienceDirect
Procedia Technology 16 (2014) 1106 – 1115
Procedia Technology

CENTERIS 2014 - Conference on ENTERprise Information Systems / ProjMAN 2014 - International Conference on Project MANagement / HCIST 2014 - International Conference on Health and Social Care Information Systems and Technologies

# Scheduling of Multimodal Activities with Multiple Renewable and Availability Constrained Resources under Stochastic Conditions

Rui Moutinho[a], Anabela Tereso[a,]*

[a]*Department of Production and Systems Engineering - CGIT Research Centre - University of Minho, 4800-058 Guimarães, Portugal*

**Abstract**

We address the stochastic multi-mode resource-constrained project scheduling problem in the case where the project activities require multiple renewable resources, constrained by total availability. In our study, we introduce uncertainty by characterizing, in each activity, each required resource with a work content following some known distribution. Furthermore, the set of feasible resource allocations is a continuous and well defined interval of real values. In turn, any allocation must also comply with the resource maximum total availability at any point during the project execution.
We propose a mathematical model that relies on a discrete and finite partition of the timespan when evaluating activity schedules and enforces equal durations yielded by each required resource, within each activity, in expectation. The objective is to determine the start times and the mode (defined by the resource allocations) which minimize the expected total project cost.
We implemented a method relying upon two global optimization metaheuristics - the electromagnetic-like mechanism and the evolutionary algorithm to solve the problem.


Peer-review under responsibility of the Organizing Committee of CENTERIS 2014.

*Keywords:* project scheduling; multimodal activities; multiple resources; renewable resources; availability constrained resources; global optimization

* Corresponding author. Tel.: 00351253510344; fax: 00351253510343.
*E-mail address:* anabelat@dps.uminho.pt


Peer-review under responsibility of the Organizing Committee of CENTERIS 2014.
doi:10.1016/j.protcy.2014.10.125

## 1. Introduction

This research relates to the stochastic resource-constrained project scheduling problem. This branch of study usually involves the minimization of the expected project makespan when the activities have a well-defined duration distribution. We refer the reader to [1] for a comprehensive overview.

The work here presented comes in succession with earlier developments [2]. The first study, devised a model introducing the uncertainty in activity durations by describing its work content as following a specified distribution. The activities, indexed by $a \in A$, were assumed *multimodal*. That is, an activity duration ($Y_a$) is to be influenced by two factors: the work content ($W_a$) and the allocated amount ($x_a$),

$$Y_a = W_a / x_a$$

That model, then, assumed all the $x_a$ to be deterministically determined, thus being the decision variables; and addresses the resource cost, where $b$ is constant (the resource unit cost):

$$C_a = b.x_a.W_a$$

Lastly, a penalty cost is taken when the actual project duration ($\Upsilon_n$) exceeds the predetermined due date ($T_s$), given by the expression $C(\Upsilon_n - T_s)$. In this first model, it was also assumed no preemption and the existence of a unique resource which is always available and infinitely abundant; shared by all the activities within a project. The objective was to determine a vector, of all the $x_a$, which minimizes the expected total cost.

The first computational approach to the model was using dynamic programming in Matlab by partitioning the activity precedence network (activity-on-arc) in the, so called, uniformly directed cutsets[†]. Despite solving the problem, such method proved to be computationally cumbersome. A second approach, still in Matlab, used a global optimization heuristic based on the electromagnetic-like mechanism [3,4] obtained promising results. The heuristic method achieved results faster than the dynamic programming one. This encouraged porting it to Java [5] and the development of another implementation using an evolutionary algorithm [6], also in Java. Both produced satisfactory results at less computational effort, mostly due to the heuristic nature of these algorithms, compared to the exact nature of dynamic programming and also due to the flexibility of the computer language (compared to Matlab).

The first model was extended to encompass the multiple resource scenario, while maintaining all other constraints and assumptions [7]. Instead of a single resource shared among the project, now each activity requires a non-empty subset of the, so called, project resources. Consequently, each resource within each activity holds a work content characterization. This implies the activity duration to be a random variable describing the maximum of each of the individual durations yielded by each one of its resources. Additionally, the model assumes that once an activity starts, any allocated quantity to any of its resources is locked until the finish (of the activity). Despite any resource being abundant, it is desired that all resources, among those requested by the same activity, are used in a way that their induced durations are equal, in expectation; as any difference would yield a waste or maintenance fee. For example, a set of batteries will decay over time even if not used; and there is no advantage trying to allocate more of one resource in order to yield minor durations if that would imply such resource to be mostly dormant, until some activity finishes. So, the model introduced the idleness cost which penalizes any observed difference on the aforementioned scenario.

In this paper, we present a model built upon the previous multiple resources model but considering the resources as *restricted* in (maximum) availability at any time and *completely renewable*. By renewable we mean that any allocated amount may be claimed entirely after the requesting activity ends. Therefore, given a project with no-preemptive multi-mode activities to which any requested resource is associated to a stochastic work content,

[†] The acronym *udc* stands for 'uniformly directed cutset', which is a cutset of the graph in which all arrows are directed from the subset of nodes H which contains the origin node, to the complementary subset $\overline{H} = N - H$ which contains the terminal node.

inferring both activity duration and allocation cost as random variables, the new model objective is to determine the resource allocations and the schedule (activity start times) as to minimize the project total cost. And, the total project cost comprises the allocation and idleness cost as well as the deviation from the predefined due date. Such complexity is our major contribution. The multiple resource case is already addressed [8,9] but, in both, the activities are single-mode and the aim is focused on the total makespan, following proactive strategies. Also, the stochastic nature of the activity durations is assumed as a well-known project parameter.

The reminder of the paper is organized as follows: new model formalization; remarks of the computational implementation; experiment results overview; discussion and finally conclusions and future research.

## 2. Restricted Multiple Renewable Resources Model

Before we can formalize the proposed model, a list of variables is provided for quick reference:

- $\boldsymbol{N} \in \mathbb{N}$ The number of project activities;
- $\boldsymbol{K} \in \mathbb{R}$ The number of project resources;
- $\mathbf{P} \subset \mathbb{N} \times \mathbb{N}$ The image set of the activity precedence relation: $(i,j) \in P \Leftrightarrow i$ precedes $j$;
- $\boldsymbol{W_{jr}} \in \mathbb{R}$ Work content of resource *r* within activity *j*;
- $\boldsymbol{\tau} \in \mathbb{R}^+$ Time unit length;
- $\boldsymbol{Y_{ir}} \in \mathbb{N}$ Individual activity duration yielded by resource *r* within activity *j*;
- $\boldsymbol{Y_i} \in \mathbb{N}$ Duration of activity *j*;
- $\boldsymbol{s_j} \in \mathbb{N}$ Start time of activity *j*;
- $\boldsymbol{x_{jr}} \in \mathbb{R}$ Allocated amount of resource *r* within activity *j*;
- $\boldsymbol{R_r} \in \mathbb{R}^+$ Maximum availability for project resource *r*;
- $\boldsymbol{c_r} \in \mathbb{R}^+$Allocation cost per unit for project resource *r*;
- $\boldsymbol{i_r} \in \mathbb{R}^+$ Idleness cost per time unit for project resource *r*;
- $T_D \in \mathbb{N}$ Project deadline - the target ceiling for the total duration;
- $c_L \in \mathbb{R}_0^+$ Cost per tardiness unit time;
- $c_E \in \mathbb{R}_0^+$ Bonus fee per earliness unit time.

The schedule $s_1, \dots, s_N$ and the allocation table $x_{11}, \dots, x_{1K}, x_{N1}, \dots, x_{NK}$ constitute the *decision variables*.

The model does not allow activity preemption but the activities may be postponed while respecting the precedence relation. The model's goal is to minimize the project total cost from a trade-off between three components: idleness cost, allocation cost and tardiness/earliness cost/bonus.

We use a discrete and finite partition of the timespan with intervals of equal length, so called $\tau$, which is taken as the time unit. This simplifies the model as it narrows the possible activity starting times, especially when postponing some activity execution. Also, it is more realistic the explicit usage of integer multiples of a pre-determined time unit - days, hours, minutes, months, etc… As an example, suppose a transportation truck that is requested and taxed by day. When only used for 2.5 days the whole 3 days would be paid.

We will refer to each period by its index in order, starting at zero. The durations become expressed as counts of contiguous periods. In figure 1, we show a representation of the absolute timespan as a sequence of time periods, for clarification.

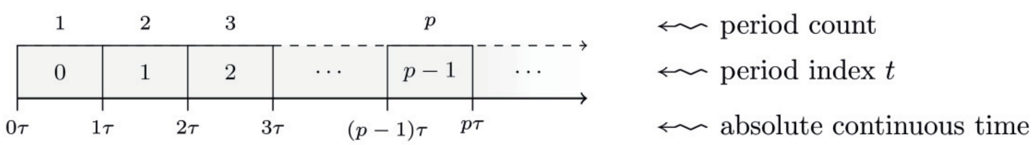


Fig. 1. Finite partition of an absolute continuous timespan.

The set of required resources by one activity is a non-empty subset of the $K$ project resources. We start by defining an auxiliary function $\delta$:

$$\delta(j,r) = \begin{cases} 1 & \text{if activity } j \text{ requires resource } k \\ 0 & \text{otherwise} \end{cases}$$

For each required resource $k$ in each activity $j$ we define its work content as a random variable, exponentially distributed with parameter $\lambda_{jk}$:

$$W_{jk} \sim Exp(\lambda_{jk}) \text{ when } \delta(j,k) = 1$$

The activities are multimodal, hence their durations will change according to the allocated resource amount. Given an allocation vector $x_{j1}, \dots, x_{jK}$ for activity $j$, the duration yielded by the resource, in time units, is determined as:

$$Y_{jk} = \eta \text{ such } \eta \times \tau = \lceil W_{jr}/x_{jk} \rceil \times \delta(j,k)$$

Notice that for each non-required resource for some activity, it is irrelevant the actual values of both work content and allocated quantity as the duration yielded by such resource always evaluates to zero (because $\delta(\mathrm{j},\mathrm{k}) = 0$) . On the other hand, for each required resource the allocated amounts (or modes) are allowed to be any real value between fixed lower and upper bounds:

$$0 < \ell_{jk} \leq x_{jk} \leq u_{jk} \leq R_k$$

From the above definitions, the activity duration is the random variable evaluated as the maximum of all individual durations supported by each of its resources:

$$Y_j = \max_{1 \leq k \leq K} Y_{jk}$$

In order to avoid wasteful idle time of any of the resources, when required to the same activity $j$, it is desired to allocate just enough of each resource to yield in equal duration, in expectation:

$$\mathcal{E}[Y_{jr}] = \mathcal{E}[Y_{js}], \forall r, s \colon \delta(j,r) = 1 \wedge \delta(j,s) = 1$$

We follow the same strategy used on the previous multiple resource model (7) by inducing the above behavior with the addition of the so called *idleness cost* (or *maintenance cost*). The idleness cost of an activity is evaluated as:

$$I_j = \sum_{1 \leq k \leq K} i_r (Y_j - Y_{jk}) \delta(j,k)$$

In order to define a schedule of activities, one must consider the activity start time. Thus, we establish the following binary variables, given an allocation vector:

$$s_{jt} = \begin{cases} 1 & \textit{if activity j starts in period } t \\ 0 & \textit{otherwise} \end{cases}$$

We do not allow preemptive activities. As such, the following constraint is applied:

$$\sum_{t \geq 0} s_{jt} = 1$$

Upon the above equation, we denote by $s_j$ the period where the activity *j* starts:

$$s_j = t \Leftrightarrow s_{jt} = 1$$

The activity start period is also constrained by the project's activity precedence relations:

$$s_i + Y_i \leq s_j, \forall\ (i, j) \in P$$

The $s_j, 1 \leq j \leq N$ vector represents a schedule of the project yielded by some allocation table.
The resources are renewable which means any allocated quantity is made available once the activity requesting it is finished. Thus, at any time period *t* the allocation table must comply with:

$$x_{jk} = 0 \Leftrightarrow t < s_j \vee t \geq s_j + Y_j, \forall\ j \in \{1, \dots, N\}, k \in \{1, \dots, K\}$$

Note that this condition implies the lower and upper bounds defined earlier be only applied within activity execution time. The resources are also restricted by their total availability. Thus, we further constrain the allocation table by:

$$\forall\ t \geq 0, 1 \leq k \leq K: \sum\nolimits_{1 \leq j \leq N} x_{jk} \leq R_k$$

Given the above conditions, we can define the total cost of the project. We denote by $C_R$ the total cost of allocation evaluated from the allocation table yielding the activity schedule:

$$C_R = \sum\nolimits_{1 \leq j \leq N} \sum\nolimits_{1 \leq k \leq K} \left(c_k \times x_{jk} \times W_{jk}\right)$$

From the schedule we evaluate the project execution time, denoted by $\mathcal{T}$, as:

$$\mathcal{T} = \max_{1 \leq j \leq N} \left(s_j + Y_j\right)$$

With the project execution time, the cost of tardiness/earliness, denoted by $C_{ET}$, is:

$$C_{ET} = (\mathcal{T} - T_D) \times \begin{cases} c_E & \mathcal{T} \leq T_D \\ c_L & \mathcal{T} > T_D \end{cases}$$

Lastly, the total idleness cost is denoted by $C_I$ and evaluated by:

$$C_I = \sum\nolimits_{1 \leq j \leq N} I_j$$

Finally, given all the conditions and definitions above, we want to minimize the total project cost, in expectation:

$$\text{minimize } \mathcal{E}[C_R + C_I + C_{ET}]$$

## 3. Computational implementation

It is noticeable, from the model, that in order to satisfy all the constraints one incurs in a circular evaluation. To determine the time periods, it is required knowledge of the resource allocations while those allocations must remain valid at all time periods.

The circularity is broken if we fix some allocations and then devise the best schedule that is yielded by such allocation table. In other words, from an allocation table we infer the activity durations, then adjust possible schedules that hold those durations while respecting the total resource availability. Once the schedule is determined, it is possible to evaluate the total cost. There is always a schedule that satisfies all the resource availability constraints: the one with all activities being started sequentially, right after all their predecessors finished. Such task is difficult due to the stochastic nature of the work contents which may lead to various durations all from the same allocation table.

From the above reasoning we chose to pursue sub-optimal solutions using heuristic global optimization techniques.

The implementation uses two interconnected heuristic cycles. The outer cycle fixes an allocation table and passes to the inner cycle the task of finding the sub-optimal (with less timespan) schedule for such allocation. Then, the total cost is evaluated and the allocation table completes a possible solution. The total cost from this solution is used to guide the outer heuristic in pursuit of the sub-optimal allocation. Since the allocation tables are real values, the electromagnetic-like mechanism was used for the outer cycle.

The inner cycle is inherently integer as the schedules consist in shifting (delaying) and swapping activities (start times) in order to maintain validation and minimum timespan. Initially, we considered to use the electromagnetic-like mechanism adaptation to the integer case [10]. It inspired us to adapt the evolutionary approach where recombination and mutation can be exploited in order to schedule the activities. In the following, we shall clarify those adaptations.

Our method, builds valid schedules by taking each activity one-by-one and determining the minimum time where it could start without violating resource nor precedence constraints. Thus, by shifting the ordering, one could possibly achieve different schedules. By taking the ordering as a vector and the obtained timespan (of the produced schedule) as a return value (solution candidate), we can represent each schedule data by an individual in the evolutionary method [11].

We applied the evolutionary strategy $(\mu/\rho + \lambda) - ES$ where: $\mu$ is the number of adult individuals able to pass information into the next generation; $\rho$ is the number of adult individuals required to produce a single new individual (child) and $\lambda$ is the number of produced individuals at each new generation (offspring); discrete recombination and non-isotropic adaptation was used. The mutation will produce a new ordering by choosing the position of the largest of the standard deviation components (associated to each component of the ordering vector) as pivot (see figure 2). Then, if the new ordering resulting from swapping that component with its immediate right neighbor is valid, the new individual is created; else it is discarded. When a new individual is about to be discarded, it is replaced by some random individual from the population. In order to contain the mutation effect small between generations, the recombination only applies to the standard deviation components; effectively, changing only the choice of the pivotal component.

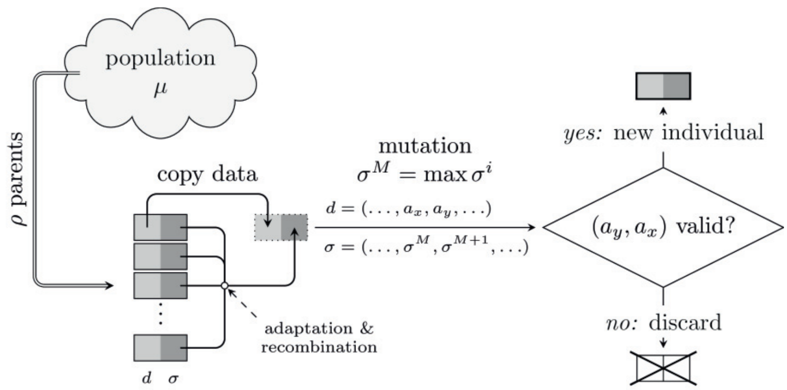


Fig. 2. Creation of a new individual: adaptation, recombination, mutation and validation.

By employing such strategy, the best schedules will be slightly changed so the sub-optimal makespan can be found. Also, by randomly choosing an existing individual to take the place of a rejected candidate (from mutation), it is given the opportunity for the method to find some local (perhaps better) solution.

## 4. Experimental results

We tested our implementation on several projects arranged in pairs where two projects share the same activity precedence relations and tardiness/earliness time unit cost/bonus but have different resource maximum availabilities with one being single-resource and the other with up to four resources.

For brevity, we only present the results from two project pairs. The relevant data of these projects may be consulted in table 1. In the reminder, we will address each project by the name of its pair followed by S for single resource or M for multiple resources. For example, A-M indicates project from pair A with multiple resources.

Table 1. Summary of the project pairs A and B whose results are presented.

| | | | | | Single resource | Multiple Resources | | |
|---|---|---|---|---|---|---|---|---|
| | $N$ | $c_L$ | $c_E$ | $T_D$ | $R_1$ | $R_1$ | $R_2$ | $R_3$ |
| A | 11 | 8 | 0.8 | 28 | 1.5 | 3 | 3 | 3 |
| B | 24 | 12 | 1.2 | 223 | 5 | 1.5 | 3 | n.a. |

Our implementation of the electromagnetic (EM) and evolutionary (EV) methods is configurable. We used two of such configurations, per each method: a *default* general setting – D – and a less cumbersome configuration – L. Table 2 and table 3 provide the settings under those configurations.

Table 2. EM configurations.

| | No. particles | Max. iterations |
|---|---|---|
| D | 10 | 250 |
| L | 3 | 75 |

Table 3. EV configurations.

| | $\mu$ | $\lambda$ | $\rho$ | Max. generations |
|---|---|---|---|---|
| D | 10 | 20 | 5 | 250 |
| L | 3 | 7 | 2 | 80 |

The experiments tackled the sensibility of the overall strategy to the different configurations. Thus, each test ran under each of the four combined configurations to each project: DD, DL, LD and LL. The nomenclature refers to the name of the configuration used for the outer cycle (EM) in the first character and the inner cycle (EV) on the other.

All tests were performed by delegating throughout a set of heterogeneous computers providing a distributed and concurrent environment for the simulations. Each test (project × configuration) ran until a minimum of 10000 results were collected. Table 4 shows the number of obtained results (#) and the mean time (in milliseconds) that each individual result took to evaluate ($\overline{\Delta}_t$) for A and B project pairs, under the four configurations.

Table 4. Simulation summary for A and B project pairs.

| | # – DD | $\overline{\Delta}_t$ – DD | # – DL | $\overline{\Delta}_t$ – DL | # – LD | $\overline{\Delta}_t$ – LD | # – LL | $\overline{\Delta}_t$ – LL |
|---|---|---|---|---|---|---|---|---|
| A-S | 119172 | 266 | 10000 | 200 | 10000 | 21 | 10000 | 8 |
| A-M | 77110 | 343 | 10000 | 86 | 10000 | 30 | 10000 | 9 |
| B-S | 47934 | 702 | 10000 | 154 | 10000 | 41 | 10000 | 13 |
| B-M | 19798 | 8107 | 10000 | 383 | 10000 | 250 | 10000 | 27 |

For our analysis of the results in terms of total cost and timespan, we evaluated the confidence intervals, at 95%, to the means of the sub-optimal costs and timespan collected. For the DD configuration the confidence interval for the timespan and total cost are shown in table 5, for project pair A and B.

Table 5. Confidence intervals, at 95%, for timespan and total cost, under DD configuration, for A and B.

| | A-S | A-M | B-S | B-M |
|---|---|---|---|---|
| Timespan (CI) | ] 58.14, 58.36 [ | ] 34.94, 35.07 [ | ] 259.189, 260.238 [ | ] 530.702, 533.468 [ |
| Total Cost (CI) | ] 406.17, 408.61 [ | ] 302.72, 304.56 [ | ] 1812.11, 1828.56 [ | ] 5955.85, 5998.64 [ |

In figure 3, a graphical representation of the confidence intervals for all four configurations, per each project, is presented. For each graphic, the horizontal axis and vertical axes represent the timespan and the total cost, respectively. The lower left corner is the point of lesser timespan and lesser cost. In order to increase readability, especially in the cases where the interval length along each axis may differ greatly, a scale is applied to each axis as to enforce a squared overall graphic shape. Note that such scales may differ between graphics.

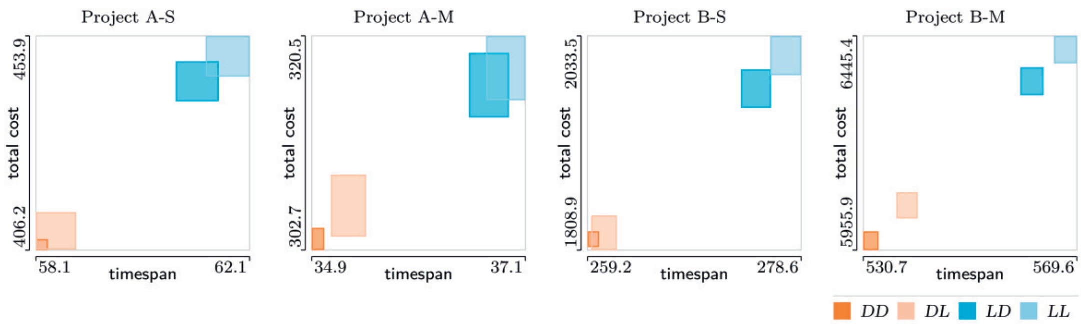


Fig. 3. Scaled representation of the confidence intervals (95%) for the timespan and total cost.

Lastly, the overall distribution of the results obtained under DD configuration can be observed in the colored scatter plots shown in figures 4 to 5. The colors (and shapes) usage allows to differentiate those results (points) with absolute frequency below percentile 90 from those above it. The most common points are marked at different color.

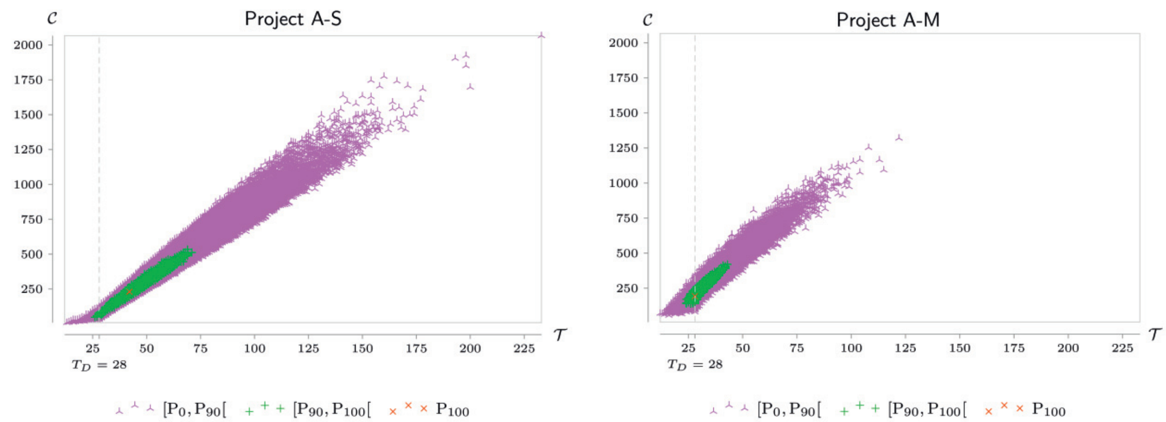


Fig. 4. Scatter plot of results collected under the DD configuration for project pair A.

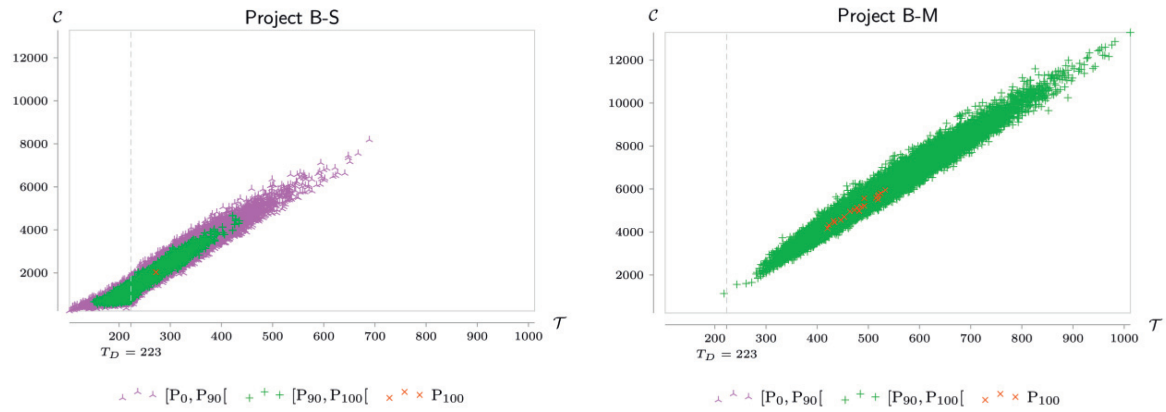


Fig. 5. Scatter plot of results collected under the DD configuration for project pair B.

## 5. Discussion

On all results the lighter configurations took much less time to execute the simulations, as expected. However, they seem not to be able to converge towards a better solution when compared with the heavier DD, as seen in figure 3. The D configuration applied to the outer cycle seems to be dominant as in the graphics both orange rectangles (intervals) are packed together on the lower left corner (less timespan and less cost) while the blue ones are in the opposite corner. Such difference is not so relevant when a heavier configuration is combined with the lighter. Again, the heavier configuration, applied to either cycle, appears to dominate over the lighter version.

Observing figure 4, for the project pair A, it is clear the effect of the bonus cost which infers a distortion around the deadline mark. Such distortion relates to two different positive correlations: one below $T_D$ and the other above it. This shows the expected behavior by which earlier project makespan is rewarded (negative cost proportional to $c_E$) while delays are penalized (positive cost proportional to $c_T$). The multiple version is thicker than the single version, suggesting more difficulty in narrowing the total cost. This is supported by the existence of the idleness cost which is being taken into account when finding the sub-optimal total cost. On both cases, the $[P_{90}, P_{100}[$ zone is interior to the $[P_0, P_{90}[$ zone and relatively small. This enforces that the algorithms tend to converge around some local solution.

Still on the scatter plots of the pair A, we can observe that the most common values are relatively near the deadline mark. In fact, in the project A-M it lays, even marginally, below that value. This may be explained by the constraints of maximum availability of the resources. On the A-S project, the maximum amount of 1.5 shows to impact greatly the timespan pushing both project execution time and total cost to higher values as the schedules become more sequential. On the other hand, the maximum availability of 3 on each of the 3 resources on the A-M project seems to allow more concurrency in the activity schedule, hence yielding lower cost and timespan.

In the case of project pair B (see figure 5), the above reasoning holds for the B-S project. As for the B-M project, the deadline occurs much lesser than the majority of the results. This infers that, perhaps, the specification of such deadline was overoptimistic. On the other hand, there are no points with absolute frequency below $P_{90}$. This suggests slow convergence to a specific point. In fact, there are many values with the maximum frequency spanning along both time and cost, which demonstrates this behavior.

## 6. Conclusions

We have devised a model to the minimization of the total cost of projects with multiple renewable resources constrained in maximum availability, under stochastic conditions and multi-mode activities. The model was implemented in Java using a combination of the electromagnetic-like mechanism and an evolutionary algorithm.

The result analysis to the achieved sub-optimal solution in terms of total cost and respective schedule/timespan seems promising. The model is sensible to both the multiplicity of resources and idleness cost while respecting its interactions with the maximum availability constraints. More results and simulations have to be made in order to fully describe the behavior of the new model and its implementation; namely, with project versions without restrictions to the resource availability.

Our approach relies mostly in repeated and self-oriented simulations which tend to converge towards better solutions implicitly as dictated by the mechanics of the heuristic methods used. Thus, previous considerations regarding priority rules or policies may not be as prevalent here as with other approaches. Such analysis, for example, should prove beneficial especially with the prediction of the cases where our method may not manifest quick convergence to a sub-optimal solution. Under these conditions, we defend that a practical application is feasible by following these guidelines, after a project is well defined:

1. Run a first set of simulations and evaluate the expected result;
2. Take the result as a baseline plan;
3. Whenever an activity is concluded, update the initial project by removing the information about such activity. Take the current project cost and duration and add it to the objective function as a constant. Return to 1.

This proposal has some caveats, which should be tackled in future research:

1. How to deal with completion of many activities in a short amount of time?
2. The rerunning of the simulations may take a long time. The overall execution of the project shall not be delayed on waiting for the updated results.

## References

[1] Demeulemeester EL, Herroelen WS. Project Scheduling - A Research Handbook. International Series in Operations Research & Management Science. 2002.

[2] Tereso AP, Araújo MMT, Elmaghraby SE. Adaptive Resource Allocation in Multimodal Activity Networks. International Journal of Production Economics. 2004 Nov; 92(1): p. 1-10.

[3] Birbil SI, Fang SC. An Electromagnetism-like Mechanism for Global Optimization. Journal of Global Optimization. 2003 Mar; 25(3): p. 263-282.

[4] Birbil SI, Fang SC, Sheu RL. On the Convergence of a Population-Based Global Optimization Algorithm. Journal of Global Optimization. 2004 Nov; 30(2): p. 301-318.

[5] Tereso AP, Novais RA, Araújo MMT. The Optimal Resource Allocation in Stochastic Activity Networks via the Electromagnetism Approach: A Platform Implementation in Java. In 21st European Conference on Operational Research (EURO XXI); 2006 July; Reykjavic, Iceland.

[6] Tereso AP, Costa LA, Novais RA, Araújo MT. The Optimal Resource Allocation in Stochastic Activity Networks via the Evolutionary Approach: A Platform Implementation in Java. In International Conference on Industrial Engineering and Systems Management; 2007 May 30 - June 2; Beijing, China.

[7] Tereso AP, Araújo MMT, Moutinho R, Elmaghraby SE. Project Management: multiple resources allocation. In International Conference on Engineering Optimization (EngOpt2008); 2008 June 1-5; Rio de Janeiro - Brazil.

[8] Bruni ME, Beraldi P, Guerriero F, Pinto E. A heuristic approach for resource constrained project scheduling with uncertain activity durations. Computers & Operations Research. 2011 Sep; 38(9).

[9] Deblaere F, Demeulemeester E, Herroelen W. Proactive policies for the stochastic resource-constrained project scheduling problem. European Journal of Operational Research. 2011 Oct; 214(2).

[10] Javadian N, Alikhani MG, Tavakkoli-Moghaddam R. A Discrete Binary Version of the Electromagnetism-Like Heuristic for Solving Traveling Salesman Problem. In Huang DS, Wunsch D, Levine D, Jo KH, editors. Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence.: Springer Berlin / Heidelberg; 2008. p. 123-130.

[11] Costa L, Oliveira P. An Evolution Strategy for Multiobjective Optimization. In IEEE Congress on Evolutionary Computation (CEC 2002); 2002; Honolulu, USA. p. 97-102.